In Proceedings of the ACL-05 Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization (MTSE'05), 2005, Ann Arbor, MI

On Some Pitfalls in Automatic Evaluation and Significance Testing for MT

Stefan Riezler and **John T. Maxwell III**
Palo Alto Research Center
3333 Coyote Hill Road, Palo Alto, CA 94304

Abstract

We investigate some pitfalls regarding the discriminatory power of MT evaluation metrics and the accuracy of statistical significance tests. In a discriminative reranking experiment for phrase-based SMT we show that the NIST metric is more sensitive than BLEU or F-score despite their incorporation of aspects of fluency or meaning adequacy into MT evaluation. In an experimental comparison of two statistical significance tests we show that p-values are estimated more conservatively by approximate randomization than by bootstrap tests, thus increasing the likelihood of type-I error for the latter. We point out a pitfall of randomly assessing significance in multiple pairwise comparisons, and conclude with a recommendation to combine NIST with approximate randomization, at more stringent rejection levels than is currently standard.

1 Introduction

Rapid and accurate detection of result differences is crucial in system development and system benchmarking. In both situations a multitude of systems or system variants has to be evaluated, so it is highly desirable to employ automatic evaluation measures for detection of result differences, and statistical hypothesis tests to assess the significance of the detected differences. When evaluating subtle differences between system variants in development, or when benchmarking multiple systems, result differences may be very small in magnitude. This imposes strong requirements on both automatic evaluation measures and statistical significance tests: Evaluation measures are needed that have high discriminative power and yet are sensitive to the interesting aspects of the evaluation task. Significance tests are required to be powerful and yet accurate, i.e., if there are significant differences they should be able to assess them, but not if there are none.

In the area of statistical machine translation (SMT), recently a combination of the BLEU evaluation metric (Papineni et al., 2002) and the bootstrap method for statistical significance testing (Efron and Tibshirani, 1993) has become popular (Och, 2003; Kumar and Byrne, 2004; Koehn, 2004b; Zhang et al., 2004). Given the current practice of reporting result differences as small as .3% in BLEU score, assessed at confidence levels as low as 70%, questions arise concerning the sensitivity of the employed evaluation metrics and the accuracy of the employed significance tests, especially when result differences are small. We believe that is important to accurately detect such small-magnitude differences in order to understand how to improve systems and technologies, even though such differences may not matter in current applications.

In this paper we will investigate some pitfalls that arise in automatic evaluation and statistical significance testing in MT research. The first pitfall concerns the discriminatory power of automatic evaluation measures. In the following, we compare the sensitivity of three intrinsic evaluation measures that differ with respect to their focus on different aspects

of translation. We consider the well-known BLEU score (Papineni et al., 2002) which emphasizes fluency by incorporating matches of high n-grams. Furthermore, we consider an F-score measure that is adapted from dependency-based parsing (Crouch et al., 2002) and sentence-condensation (Riezler et al., 2003). This measure matches grammatical dependency relations of parses for system output and reference translations, and thus emphasizes semantic aspects of translational adequacy. As a third measure we consider NIST (Doddington, 2002), which favors lexical choice over word order and does not take structural information into account. On an experimental evaluation on a reranking experiment we found that only NIST was sensitive enough to detect small result differences, whereas BLEU and F-score produced result differences that were statistically not significant. A second pitfall addressed in this paper concerns the relation of power and accuracy of significance tests. In situations where the employed evaluation measure produces small result differences, the most powerful significance test is demanded to assess statistical significance of the results. However, accuracy of the assessments of significance is seldom questioned. In the following, we will take a closer look at the bootstrap test and compare it with the related technique of approximate randomization (Noreen (1989)). In an experimental evaluation on our reranking data we found that approximate randomization estimated p-values more conservatively than the bootstrap, thus increasing the likelihood of type-I error for the latter test. Lastly, we point out a common mistake of randomly assessing significance in multiple pairwise comparisons (Cohen, 1995). This is especially relevant in k-fold pairwise comparisons of systems or system variants where k is high. Taking this multiplicity problem into account, we conclude with a recommendation of a combination of NIST for evaluation and the approximate randomization test for significance testing, at more stringent rejection levels than is currently standard in the MT literature. This is especially important in situations where multiple pairwise comparisons are conducted, and small result differences are expected.

2 The Experimental Setup: Discriminative Reranking for Phrase-Based SMT

The experimental setup we employed to compare evaluation measures and significance tests is a discriminative reranking experiment on 1000-best lists of a phrase-based SMT system. Our system is a re-implementation of the phrase-based system described in Koehn (2003), and uses publicly available components for word alignment (Och and Ney, 2003)[1], decoding (Koehn, 2004a)[2], language modeling (Stolcke, 2002)[3] and finite-state processing (Knight and Al-Onaizan, 1999)[4]. Training and test data are taken from the Europarl parallel corpus (Koehn, 2002)[5].

Phrase-extraction follows Och et al. (1999) and was implemented by the authors: First, the word aligner is applied in both translation directions, and the intersection of the alignment matrices is built. Then, the alignment is extended by adding immediately adjacent alignment points and alignment points that align previously unaligned words. From this many-to-many alignment matrix, phrases are extracted according to a contiguity requirement that states that words in the source phrase are aligned only with words in the target phrase, and vice versa.

Discriminative reranking on a 1000-best list of translations of the SMT system uses an ℓ_1 regularized log-linear model that combines a standard maximum-entropy estimator with an efficient, incremental feature selection technique for ℓ_1 regularization (Riezler and Vasserman, 2004). Training data are defined as pairs $\{(s_j, t_j)\}_{j=1}^{m}$ of source sentences s_j and gold-standard translations t_j that are determined as the translations in the 1000-best list that best match a given reference translation. The objective function to be minimized is the conditional log-likelihood $L(\boldsymbol{\lambda})$ subject to a regularization term $R(\boldsymbol{\lambda})$, where $T(s)$ is the set of 1000-best translations for sentence s, $\boldsymbol{\lambda}$ is a vector or log-parameters, and

[1] http://www.fjoch.com/GIZA++.html
[2] http://www.isi.edu/licensed-sw/pharaoh/
[3] http://www.speech.sri.com/projects/srilm/
[4] http://www.isi.edu/licensed-sw/carmel/
[5] http://people.csail.mit.edu/people/koehn/publications/europarl/

Table 1: NIST, BLEU, F-scores for reranker and baseline on development set

	NIST	BLEU	F
baseline	6.43	.301	.385
reranking	6.58	.298	.383
approxrand p-value	$< .0001$	.158	.424
bootstrap p-value	$< .0001$	.1	-

$\boldsymbol{f}$ is a vector of feature functions:

$$\begin{aligned} L(\boldsymbol{\lambda}) + R(\boldsymbol{\lambda}) &= -\log \prod_{j=1}^{m} p_{\boldsymbol{\lambda}}(t_j|s_j) + R(\boldsymbol{\lambda}) \\ &= -\sum_{j=1}^{m} \log \frac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(t_j)}}{\sum_{t\in T(s_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(t)}} + R(\boldsymbol{\lambda}) \end{aligned}$$

The features employed in our experiments consist of 8 features corresponding to system components (distortion model, language model, phrase-translations, lexical weights, phrase penalty, word penalty) as provided by PHARAOH, together with a multitude of overlapping phrase features. For example, for a phrase-table of phrases consisting of maximally 3 words, we allow all 3-word phrases and 2-word phrases as features. Since bigram features can overlap, information about trigrams can be gathered by composing bigram features even if the actual trigram is not seen in the training data.

Feature selection makes it possible to employ and evaluate a large number of features, without concerns about redundant or irrelevant features hampering generalization performance. The ℓ_1 regularizer is defined by the weighted ℓ_1-norm of the parameters

$$R(\boldsymbol{\lambda}) = \gamma||\boldsymbol{\lambda}||_1 = \gamma \sum_{i=1}^{n} |\lambda_i|$$

where γ is a regularization coefficient, and n is number of parameters. This regularizer penalizes overly large parameter values in their absolute values, and tends to force a subset of the parameters to be exactly zero at the optimum. This fact leads to a natural integration of regularization into incremental feature selection as follows: Assuming a tendency of the ℓ_1 regularizer to produce a large number of zero-valued parameters at the function's optimum, we start with all-zero weights, and incrementally add features to the model only if adjusting their parameters away from zero sufficiently decreases the optimization criterion. Since every non-zero weight added to the model incurs a regularizer penalty of $\gamma|\lambda_i|$, it only makes sense to add a feature to the model if this penalty is outweighed by the reduction in negative log-likelihood. Thus features considered for selection have to pass the following test:

$$\left|\frac{\partial L(\boldsymbol{\lambda})}{\partial \lambda_i}\right| > \gamma$$

This gradient test is applied to each feature and at each step the features that pass the test with maximum magnitude are added to the model. This provides both efficient and accurate estimation with large feature sets.

Work on discriminative reranking has been reported before by Och and Ney (2002), Och et al. (2004), and Shen et al. (2004). The main purpose of our reranking experiments is to have a system that can easily be adjusted to yield system variants that differ at controllable amounts. For quick experimental turnaround we selected the training and test data from sentences with 5 to 15 words, resulting in a training set of 160,000 sentences, and a development set of 2,000 sentences. The phrase-table employed was restricted to phrases of maximally 3 words, resulting in 200,000 phrases.

3 Detecting Small Result Differences by Intrinsic Evaluations Metrics

The intrinsic evaluation measures used in our experiments are the well-known BLEU (Papineni et al., 2002) and NIST (Doddington, 2002) metrics, and an F-score measure that adapts evaluation techniques from dependency-based parsing (Crouch et al., 2002) and sentence-condensation (Riezler et al., 2003) to machine translation. All of these measures

Set $c = 0$
Compute statistic of score differences $S_{\mathrm{X}} - S_{\mathrm{Y}}$ on test data
For random shuffles $r = 1, \ldots, R$
    For examples in test set
        Shuffle variable tuples between systems X and Y with probability 0.5
    Compute pseudo-statistic $S_{\mathrm{X_r}} - S_{\mathrm{Y_r}}$ on shuffled data
    If $|S_{\mathrm{X_r}} - S_{\mathrm{Y_r}}| \geq |S_{\mathrm{X}} - S_{\mathrm{Y}}|$
        $c++$
$p = c/R$
Reject null hypothesis if p is less than or equal to specified rejection level α.

Figure 1: Approximate Randomization Test for Statistical Significance Testing

evaluate document similarity of SMT output against manually created reference translations. The measures differ in their focus on different entities in matching, corresponding to a focus on different aspects of translation quality.

BLEU and NIST both consider n-grams in source and reference strings as matching entities. BLEU weighs all n-grams equally whereas NIST puts more weight on n-grams that are more informative, i.e., occur less frequently. This results in BLEU favoring matches in larger n-grams, corresponding to giving more credit to correct word order. NIST weighs lower n-grams more highly, thus it gives more credit to correct lexical choice than to word order.

F-score is computed by parsing reference sentences and SMT outputs, and matching grammatical dependency relations. The reported value is the harmonic mean of precision and recall, which is defined as $(2 \times \text{ precision } \times \text{ recall })/(\text{ precision } + \text{ recall })$. Precision is the ratio of matching dependency relations to the total number of dependency relations in the parse for the system translation, and recall is the ratio of matches to the total number of dependency relations in the parse for the reference translation. The goal of this measure is to focus on aspects of meaning in measuring similarity of system translations to reference translations, and to allow for meaning-preserving word order variation.

Evaluation results for a comparison of reranking against a baseline model that only includes features corresponding to the 8 system components are shown in Table 1. Since the task is a comparison of system variants for development, all results are reported on the development set of 2,000 examples of length 5-15. The reranking model achieves an increase in NIST score of .15 units, whereas BLEU and F-score decrease by .3% and .2% respectively. However, as measured by the statistical significance tests described below, the differences in BLEU and F-scores are not statistically significant with p-values exceeding the standard rejection level of $.05$. In contrast, the differences in NIST score are highly significant. These findings correspond to results reported in Zhang et al. (2004) showing a higher sensitivity of NIST versus BLEU to small result differences. Taking also the results from F-score matching in account, we can conclude that similarity measures that are based on matching more complex entities (such as BLEU's higher n-grams or F's grammatical relations) are not as sensitive to small result differences as scoring techniques that are able to distinguish models by matching simpler entities (such as NIST's focus on lexical choice). Furthermore, we get an indication that differences of .3% in BLEU score or .2% in F-score might not be large enough to conclude statistical significance of result differences. This leads to questions of power and accuracy of the employed statistical significance tests which will be addressed in the next section.

4 Assessing Statistical Significance of Small Result Differences

The bootstrap method is an example for a computer-intensive statistical hypothesis test (see, e.g., Noreen (1989)). Such tests are designed to assess result differences with respect to a test statistic in cases where the sampling distribution of the test statistic

Set $c = 0$
Compute statistic of score differences $S_{\mathrm{X}} - S_{\mathrm{Y}}$ on test data
Calculate mean $\tau_B = \frac{1}{B} \sum_{b=1}^{B} S_{\mathrm{X_b}} - S_{\mathrm{Y_b}}$ over bootstrap samples $b = 1, \ldots, B$
For bootstrap samples $b = 1, \ldots, B$
    Sample with replacement from variable tuples for systems X and Y
    Compute pseudo-statistic $S_{\mathrm{X_b}} - S_{\mathrm{Y_b}}$ on bootstrap data
    If $|S_{\mathrm{X_b}} - S_{\mathrm{Y_b}} - \tau_B| \geq |S_{\mathrm{X}} - S_{\mathrm{Y}}|$
      $c++$
$p = c/B$
Reject null hypothesis if p is less than or equal to specified rejection level α.

Figure 2: Bootstrap Test for Statistical Significance Testing

is unknown. Comparative evaluations of outputs of SMT systems according to test statistics such as differences in BLEU, NIST, or F-score are examples of this situation. The attractiveness of computer-intensive significance tests such as the bootstrap or the approximate randomization method lies in their power and simplicity. As noted in standard textbooks such as Cohen (1995) or Noreen (1989) such tests are as powerful as parametric tests when parametric assumptions are met and they outperform them when parametric assumptions are violated. Because of their generality and simplicity they are also attractive alternatives to conventional non-parametric tests (see, e.g., Siegel (1988)). The power of these tests lies in the fact that they answer only a very simple question without making too many assumptions that may not be met in the experimental situation. In case of the approximate randomization test, only the question whether two samples are related to each other is answered, without assuming that the samples are representative of the populations from which they were drawn. The bootstrap method makes exactly this one assumption. This makes it formally possible to draw inferences about population parameters for the bootstrap, but not for approximate randomization. However, if the goal is to assess statistical significance of a result difference between two systems the approximate randomization test provides the desired power and accuracy whereas the bootstrap's advantage to draw inferences about population parameters comes at the price of reduced accuracy. Noreen summarizes this shortcoming of the bootstrap technique as follows: "The principal disadvantage of [the bootstrap] method is that the null hypothesis may be rejected because the shape of the sampling distribution is not well-approximated by the shape of the bootstrap sampling distribution rather than because the expected value of the test statistic differs from the value that is hypothesized."(Noreen (1989), p. 89). Below we describe these two test procedures in more detail, and compare them in our experimental setup.

4.1 Approximate Randomization

An excellent introduction to the approximate randomization test is Noreen (1989). Applications of this test to natural language processing problems can be found in Chinchor et al. (1993).

In our case of assessing statistical significance of result differences between SMT systems, the test statistic of interest is the absolute value of the difference in BLEU, NIST, or F-scores produced by two systems on the same test set. These test statistics are computed by accumulating certain count variables over the sentences in the test set. For example, in case of BLEU and NIST, variables for the length of reference translations and system translations, and for n-gram matches and n-gram counts are accumulated over the test corpus. In case of F-score, variable tuples consisting of the number of dependency-relations in the parse for the system translation, the number of dependency-relations in the parse for the reference translation, and the number of matching dependency-relations between system and reference parse, are accumulated over the test set.

Under the null hypothesis, the compared systems are not different, thus any variable tuple produced by one of the systems could have been produced just as

Table 2: NIST scores for equivalent systems under bootstrap and approximate randomization tests.

compared systems	1:2	1:3	1:4	1:5	1:6
NIST difference	.031	.032	.029	.028	.036
approxrand p-value	.03	.025	.05	.079	.028
bootstrap p-value	.014	.013	.028	.039	.013

likely by the other system. So shuffling the variable tuples between the two systems with equal probability, and recomputing the test statistic, creates an approximate distribution of the test statistic under the null hypothesis. For a test set of S sentences there are 2^S different ways to shuffle the variable tuples between the two systems. Approximate randomization produce shuffles by random assignments instead of evaluating all 2^S possible assignments. Significance levels are computed as the percentage of trials where the pseudo statistic, i.e., the test statistic computed on the shuffled data, is greater than or equal to the actual statistic, i.e., the test statistic computed on the test data. A sketch of an algorithm for approximate randomization testing is given in Fig. 1.

4.2 The Bootstrap

An excellent introduction to the technique is the textbook by Efron and Tibshirani (1993). In contrast to approximate randomization, the bootstrap method makes the assumption that the sample is a representative "proxy" for the population. The shape of the sampling distribution is estimated by repeatedly sampling (with replacement) from the sample itself.

A sketch of a procedure for bootstrap testing is given in Fig. 2. First, the test statistic is computed on the test data. Then, the sample mean of the pseudo statistic is computed on the bootstrapped data, i.e., the test statistic is computed on bootstrap samples of equal size and averaged over bootstrap samples.

In order to compute significance levels based on the bootstrap sampling distribution, we employ the "shift" method described in Noreen (1989). Here it is assumed that the sampling distribution of the null hypothesis and the bootstrap sampling distribution have the same shape but a different location. The location of the bootstrap sampling distribution is shifted so that it is centered over the location where the null hypothesis sampling distribution should be centered. This is achieved by subtracting from each value of the pseudo-statistic its expected value τ_B and then adding back the expected value τ of the test statistic under the null hypothesis. τ_B can be estimated by the sample mean of the bootstrap samples; τ is 0 under the null hypothesis. Then, similar to the approximate randomization test, significance levels are computed as the percentage of trials where the (shifted) pseudo statistic is greater than or equal to the actual statistic.

4.3 Power vs. Type I Errors

In order to evaluate accuracy of the bootstrap and the approximate randomization test, we conduct an experimental evaluation of type-I errors of both bootstrap and approximate randomization on real data. Type-I errors indicate failures to reject the null hypothesis when it is true. We construct SMT system variants that are essentially equal but produce superficially different results. This can be achieved by constructing reranking variants that differ in the redundant features that are included in the models, but are similar in the number and kind of selected features. The results of this experiment are shown in Table 2. System 1 does not include irrelevant features, whereas systems 2-6 were constructed by adding a slightly different number of features in each step, but resulted in the same number of selected features. Thus competing features bearing the same information are exchanged in different models, yet overall the same information is conveyed by slightly different feature sets. The results of Table 2 show that the bootstrap method yields p-values $< .015$ in 3 out of 5 pairwise comparisons whereas the approximate randomization test yields p-values $\geq .025$ in all cases. Even if the true p-value is unknown, we can say that the approximate randomization test estimates p-values more conservatively than the bootstrap, thus increasing the likelihood of type-I error for the bootstrap test. For a restrictive significance level of 0.15, which is motivated below for multiple

comparisons, the bootstrap would assess statistical significance in 3 out of 5 cases whereas statistical significance would not be assessed under approximate randomization. Assuming equivalence of the compared system variants, these assessments would count as type-I errors.

4.4 The Multiplicity Problem

In the experiment on type-I error described above, a more stringent rejection level than the usual .05 was assumed. This was necessary to circumvent a common pitfall in significance testing for k-fold pairwise comparisons. Following the argumentation given in Cohen (1995), the probability of randomly assessing statistical significance for result differences in k-fold pairwise comparisons grows exponentially in k. Recall that for a pairwise comparison of systems, specifying that $p < .05$ means that the probability of incorrectly rejecting the null hypothesis that the systems are not different be less than .05. Caution has to be exercised in k-fold pairwise comparisons: For a probability p_c of incorrectly rejecting the null hypothesis in a specific pairwise comparison, the probability p_e of at least once incorrectly rejecting this null hypothesis in an experiment involving k pairwise comparisons is

$$p_e \approx 1 - (1 - p_c)^k$$

For large values of k, the probability of concluding result differences incorrectly at least once is undesirably high. For example, in benchmark testing of 15 systems, $15(15-1)/2 = 105$ pairwise comparisons will have to be conducted. At a per-comparison rejection level $p_c = .05$ this results in an experimentwise error $p_e = .9954$, i.e., the probability of at least one spurious assessment of significance is $1 - (1 - .05)^{105} = .9954$. One possibility to reduce the likelihood that one ore more of differences assessed in pairwise comparisons is spurious is to run the comparisons at a more stringent per-comparison rejection level. Reducing the per-comparison rejection level p_c until an experimentwise error rate p_e of a standard value, e.g., .05, is achieved, will favor p_e over p_c. In the example of 5 pairwise comparisons described above, a per-comparison error rate $p_c = .015$ was sufficient to achieve an experimentwise error rate $p_e \approx .07$. In many cases this technique would require to reduce p_c to the point where a result difference has to be unrealistically large to be significant. Here conventional tests for post-hoc comparisons such as the Scheffé or Tukey test have to be employed (see Cohen (1995), p. 185ff.).

5 Conclusion

Situations where a researcher has to deal with subtle differences between systems are common in system development and large benchmark tests. We have shown that it is useful in such situations to trade in expressivity of evaluation measures for sensitivity. For MT evaluation this means that recording differences in lexical choice by the NIST measure is more useful than failing to record differences by employing measures such as BLEU or F-score that incorporate aspects of fluency and meaning adequacy into MT evaluation. Similarly, in significance testing, it is useful to trade in the possibility to draw inferences about the sampling distribution for accuracy and power of the test method. We found experimental evidence confirming textbook knowledge about reduced accuracy of the bootstrap test compared to the approximate randomization test. Lastly, we pointed out a well-known problem of randomly assessing significance in multiple pairwise comparisons. Taking these findings together, we recommend for multiple comparisons of subtle differences to combine the NIST score for evaluation with the approximate randomization test for significance testing, at more stringent rejection levels than is currently standard in the MT literature.

References

Nancy Chinchor, Lynette Hirschman, and David D. Lewis. 1993. Evaluating message understanding systems: An analysis of the third message understanding conference (MUC-3). *Computational Linguistics*, 19(3):409–449.

Paul R. Cohen. 1995. *Empirical Methods for Artificial Intelligence*. The MIT Press, Cambridge, MA.

Richard Crouch, Ronald M. Kaplan, Tracy H. King, and Stefan Riezler. 2002. A comparison of evaluation metrics for a broad-coverage stochastic parser. In *Proceedings of the "Beyond PARSEVAL" Workshop at the 3rd International Conference on Language Resources and Evaluation (LREC'02)*, Las Palmas, Spain.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence

statistics. In *Proceedings of the ARPA Workshop on Human Language Technology*.

Bradley Efron and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman and Hall, New York.

Kevin Knight and Yaser Al-Onaizan. 1999. A primer on finite-state software for natural language processing. Technical report, USC Information Sciences Institute, Marina del Rey, CA.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology Conference and the 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*, Edmonton, Cananda.

Philipp Koehn. 2002. Europarl: A multilingual corpus for evaluation of machine translation. Technical report, USC Information Sciences Institute, Marina del Rey, CA.

Philipp Koehn. 2004a. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Proceedings of the 6th Conference of the Association for Machine Translation in the Americas (AMTA'04)*, Washington, DC.

Philipp Koehn. 2004b. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP'04)*, Barcelona, Spain.

Shankar Kumar and William Byrne. 2004. Minimum Bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL'04)*, Boston, MA.

Eric W. Noreen. 1989. *Computer Intensive Methods for Testing Hypotheses. An Introduction*. Wiley, New York.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, PA.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. In *Proceedings of the 1999 Conference on Empirical Methods in Natural Language Processing (EMNLP'99)*, College Park, MD.

Franz Josef Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Ketherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. A smorgasbord of features for statistical machine translation. In *Proceedings of the Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL'04)*, Boston, MA.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the Human Language Technology Conference and the 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*, Edmonton, Cananda.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL)*, Philadelphia, PA.

Stefan Riezler and Alexander Vasserman. 2004. Incremental feature selection and ℓ_1 regularization for relaxed maximum-entropy modeling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP'04)*, Barcelona, Spain.

Stefan Riezler, Tracy H. King, Richard Crouch, and Annie Zaenen. 2003. Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for lexical-functional grammar. In *Proceedings of the Human Language Technology Conference and the 3rd Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*, Edmonton, Cananda.

Libin Shen, Anoop Sarkar, and Franz Josef Och. 2004. Discriminative reranking for machine translation. In *Proceedings of the Human Language Technology conference / North American chapter of the Association for Computational Linguistics annual meeting (HLT/NAACL'04)*, Boston, MA.

Sidney Siegel and John Castellan. 1988. *Nonparametric Statistics for the Behavioral Sciences. Second Edition*. MacGraw-Hill, Boston, MA.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, Denver, CO.

Ying Zhang, Stephan Vogel, and Alex Waibel. 2004. Interpreting BLEU/NIST scores: How much improvement do we need to have a better system? In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal.